Solution sketches for Assignment 4

Problem 1. (a) As we saw in class, a line ℓ intersects a triangle Δabc iff the dual point ℓ^* lies below the upper envelope U_Δ and above the lower envelope L_Δ of the lines a^*, b^*, c^* dual to the vertices of Δ. Denote by K_Δ the region between U_Δ and L_Δ.

Hence, a line ℓ crosses all the given triangles iff ℓ^* lies in the intersection of all the regions K_Δ. For this, ℓ^* has to lie below all the upper envelopes U_Δ and above all the lower envelopes L_Δ. That is, ℓ^* has to lie below the lower envelope of the upper envelopes U_Δ and above the upper envelope of the lower envelopes L_Δ. Each envelope U_Δ, L_Δ consists of two rays or of two rays and a segment, so their envelope is also a sequence of pieces of these rays and segments, similar to what's drawn in the figure.

(b) Proceed as in the hint, and note that merging two sub-envelopes, i.e., computing their upper or lower envelope, can be done in time linear in their complexities, by a trivial special case of line sweeping. Using the obvious Divide-and-Conquer approach, of partitioning the set of envelopes into two subsets of half the size, computing recursively the upper or lower envelope of each subset, and merging the resulting envelopes, as above, leads to an overall algorithm with $O(n \log^2 n)$ time (using the bound on the complexity of an envelope, as promised in the problem).

(c) In the primal plane, a line misses all the triangles (in this special configuration) iff it misses their convex hull. So we simply compute their hull, in $O(n \log n)$ time, and get the desired M using the first duality. (Details omitted.)

In the dual plane, using the first duality, a line ℓ misses all the triangles iff all their vertices lie on the same side of ℓ as the origin, which happens iff ℓ^* and the origin lie on the same side of all the dual lines to the vertices. That is, ℓ^* lies in the intersection of the halfplanes of the dual lines that contain the origin.

Problem 2. It is easy to determine whether there exists a vertical separating line, so ignore this case. Write the equation of the separating line ℓ as $y = ax+b$. Handle separately the two cases where A should be above ℓ and B below, and where A should be below ℓ and B above. For the first case, for each $p_i = (\xi_i, \eta_i) \in A$, we require that $\eta_i \geq a\xi_i + b$, and for each $q_j = (\xi'_j, \eta'_j) \in A$, we require that $\eta'_j \leq a\xi'_j + b$. These are $m + n$ linear inequalities in a and b, so we check whether this linear programming problem is feasible. (We can "invent" any objective function; for example, by maximizing a, we find the separating line with maximum slope, etc.)

Problem 3. Let (x, y) be the center of the circle, and let r be its radius. The distance of (x, y) from the line $a_i x + b_i y + c = 0$ is $\dfrac{|a_i x + b_i y + c|}{\sqrt{a_i^2 + b_i^2}}$. We can assume that $a_i^2 + b_i^2 = 1$ for each i. Also, we require that $a_i x + b_i y + c \geq 0$, so we can drop the absolute value. In short, we have the inequalities

$$a_i x + b_i y + c \geq r,$$

and $r \geq 0$, and we want to maximize r. This is an LP problem in three variables, and we solve it in $O(n)$ time. (We can drop the constraint $r \geq 0$: If the maximum r is negative, the problem is infeasible, in the sense that the intersection of the halfplanes is empty.)

Problem 4. Follow the hint: Fix the two points a, b, and vary the third vertex c. It is easy to show that the area of Δabc, with a, b fixed, is proportional to the *vertical* distance from c to the line ℓ through a and b.

Write the equation of ℓ as $y = px + q$. Then this vertical distance is $|c_y - pc_x - q|$. Notice that this expression is preserved when passing to the dual plane (using the second duality), because ℓ becomes the point $(p, -q)$, and c becomes the line $y = c_x x - c_y$, and the vertical distance is then $|-q - pc_x + c_y|$, which is the same.

So we move to the dual plane, get a collection of n lines, and run a sweeping algorithm. Every time we encounter an intersection point v of two lines a^* and b^* (the dual to the line through a and b), we find the two dual lines directly above and below v — these are the candidates for the third vertex c — and we compute the area of the corresponding primal triangle Δabc. We return the smallest-area triangle.

For the case where we want to test whether there exist three collinear points, this will be detected when we encounter an intersection point with more than two lines through it.